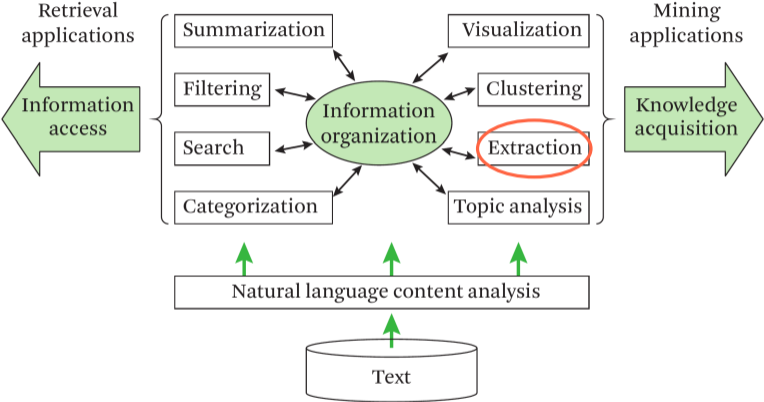


Information Extraction

Marcel Bollmann

Department of Computer and Information Science (IDA)

Reminder: Conceptual framework



Zhai and Massung (2016)

Outline

1. Introduction

2. Named Entity Recognition

- Entity Types
- Challenges
- Models
- Evaluation

3. Knowledge Bases

- WordNet
- ConceptNet
- DBpedia
- Entity Linking

4. Relation Extraction

- Regular Expressions
- Dependency Parsing

What is Information Extraction?



Information extraction

Definition

Information extraction (IE) is the task of extracting **structured** information from text.

- **Named entities**
 - e.g. persons, organisations
- **Semantic relations** between entities
 - e.g. X is-leader-of Y



“Who murdered the victim, where did it happen, and which weapon was used?”

Example: Information extraction for disease outbreak reports

As of 15 Mar 2002, Hawaii state health officials reported one additional recent case of dengue fever and 6 cases that occurred last year but were not confirmed by laboratory testing until 2002.

attribute	value
docno	ProMed.20020322.11
doc_date	2002.03.22
disease_name	dengue fever
norm_stime	2002.03.15
norm_etime	2002.03.15
victim_types	—
location	Hawaii

Source: [Grishman et al. \(2002\)](#)

Example: DBpedia

The Lord of the Rings

property	value
dbo:author	dbr:J. R. R. Tolkien
dbo:literaryGenre	dbr:Adventure novel dbr:High fantasy
dbo:previousWork	dbr:The Hobbit
dbo:publicationDate	1954-07-29
dbo:publisher	dbr:Allen & Unwin
dbo:wikiPageID	29798
dbp:language	English

Source: [DBpedia](#)

Use cases

- IE can help **find structured information** expressed in natural language.
 - e.g. company acquisitions: X bought Y
- IE can be used to **create (or update) knowledge bases**.
 - knowledge graphs, DBpedia
- IE can support **question answering** systems.
 - e.g. IBM's Watson

Named Entity Recognition



Named entity recognition

Definition

Named entity recognition (NER) is the task of **finding mentions** of named entities in a text, and **labelling them** with their type.

- Some commonly used named entity types:
 1. Person (PER)
 2. Location (LOC)
 3. Organization (ORG)

Named entity recognition: Example

Citing high fuel prices, **United Airlines**_{ORG} said **Friday**_{TIME} it has increased fares by **\$6**_{MONEY} per round trip on flight to some cities also served by lower-cost carriers. **American Airlines**_{ORG}, a unit of **AMR Corp.**_{ORG}, immediately matched the move, spokesman **Tim Wagner**_{PER} said. **United**_{ORG}, a unit of **UAL Corp.**_{ORG}, said the increase took effect **Thursday**_{TIME} and applies to most routes where it competes against discount carriers, such as **Chicago**_{LOC} to **Dallas**_{LOC} and **Denver**_{LOC} to **San Francisco**_{LOC}.

Adapted from [Jurafsky & Martin \(2023\)](#)

Properties of named entities

- Can be referred to with a proper name
 - e.g. United Airlines, Tim Wagner
- Can be indexed and linked to
 - e.g. United Airlines → [Q174769](#) (Wikidata ID)
- Participate in semantic relations
 - e.g. Tim Wagner is-spokesman-of American Airlines
- Are common answers in question answering systems
 - e.g. When did United Airlines say that it increased fares by \$6? → Friday

Entity types in OntoNotes 5

type	description
person	People, including fictional
norp	nationalities or religious or political groups
facility	Buildings, airports, highways, bridges, etc.
organization	Companies, agencies, institutions, etc.
gpe	Countries, cities, states (<i>“Geo-political entity”</i>)
location	Non-GPE locations, mountain ranges, bodies of water
product	Vehicles, weapons, foods, etc. (not services)
event	Named hurricanes, battles, wars, sports events, etc.
work of art	Titles of books, songs, etc.
law	Named documents made into laws
language	Any named language

Source: [OntoNotes Release 5.0](#)

Entity types in WNUT2017

type	description
person	People, including fictional
location	Locations, incl. geo-political entities, facilities
corporation	Corporations, businesses
product	Consumer products, tangible goods
creative-work	Titles of songs, movies, books, etc.
group	Music bands, sports teams, non-corporate organisations, etc.

Source: [Derczynski et al. \(2017\)](#)

Challenges: Type ambiguities

- Washington_{PER} was born into slavery.
- Washington_{ORG} went up 2 games to 1 in the four-game series.
- Blair arrived in Washington_{LOC} for his last state visit.
- In June, Washington_{GPE} passed a primary seatbelt law.
- The Washington_{VEH} had proved to be a leaky ship, ...

Adapted from Jurafsky & Martin (2023)

Challenges: Inflected word forms

case	inflected form
nominative	Muammar Kaddafi
genitive	Muammara Kaddafiego
dative	Muammarowi Kaddafiemu
accusative	Muammara Kaddafiego
instrumental	Muammarem Kaddafim
locative	Muammarze Kaddafim
vocative	Muammarze Kaddafi

Example: Inflected names in Polish

Source: Piskorski & Yangarber (2012)

Named entity recognition as sequence labelling

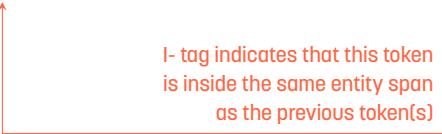
- NER can be modelled as a word-by-word **sequence labelling** task.
- We need to encode two properties for each named entity:
 1. Boundaries (*i.e.*, *which words belong to the mentioned entity*)
 2. Entity type
- A common encoding is the **BIO scheme**, where there is a tag for:
 - tokens **beginning (B)** an entity span of a given type
 - tokens **inside (I)** an entity span of a given type
 - tokens **outside (O)** of any entity

Named entity recognition: BIO scheme




American Airlines_{ORG} immediately matched the move Wagner_{PER} said

American	Airlines	immediately	matched	the	move	Wagner	said
B-ORG	I-ORG	0	0	0	0	B-PER	0

I- tag indicates that this token
is inside the same entity span
as the previous token(s)



Models for named entity recognition

- Any **sequence labelling model** can be used for named entity recognition.
 - e.g. hidden Markov model (HMM), linear-chain conditional random field (CRF)
- State-of-the-art NER models use **neural architectures**.
 -  **Stanza**: bi-directional LSTM + CRF
 -  **SparkNLP**: character-based CNN + bi-directional LSTM + CRF
 -  **spaCy**: fine-tuned RoBERTa model

Gazetteers

- Outside of text mining, a **gazetteer** is “a geographical index or dictionary.”
(Source: Oxford Languages)
- In the context of NER, it refers to a **list of named entities**.
- Gazetteers can be used as an **additional data source** to inform named entity recognizers.

Andorra	Aruba
United Arab Emirates	Azerbaijan
Afghanistan	Bosnia and Herzegovina
Antigua and Barbuda	Barbados
Anguilla	Bangladesh
Albania	Belgium
Armenia	Burkina Faso
Angola	Bulgaria
Argentina	Bahrain
American Samoa	Burundi
Austria	Benin
Australia	...

Example: List of country names

Evaluating named entity recognition

- Named entities can be viewed as **spans over tokens** annotated with their type.
- We can represent them as **tuples** containing:
 1. the **start position** of the span
 2. the **end position** of the span
 3. the **entity type**
- We can then compute **precision, recall, F1-score** based on these tuples.

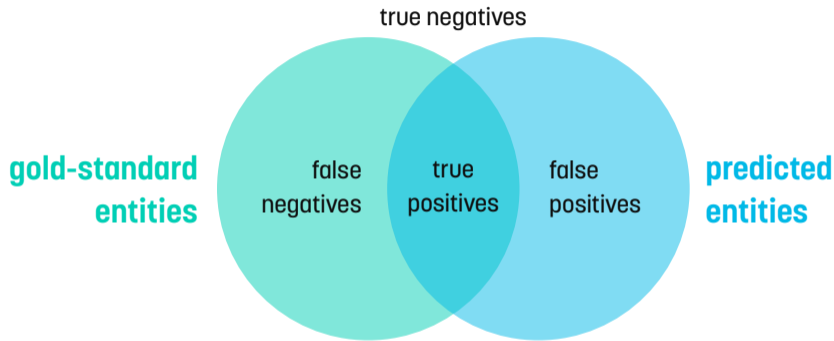
Named entities as spans

1	2	3	4	5	6	7
Foreign	ministry	spokesman	Shen	Guofang	told	Reuters
B-ORG	I-ORG	0	B-PER	I-PER	0	B-ORG

- Corresponding entity spans: (1, 2, ORG)
(4, 5, PER)
(7, 7, ORG)

Example from CoNLL 2003 NER data

Precision and recall for named entity recognition



$$P = \frac{|\text{gold} \cap \text{predicted}|}{|\text{predicted}|}$$

$$R = \frac{|\text{gold} \cap \text{predicted}|}{|\text{gold}|}$$

Precision and recall: Example

	1	2	3	4	5	6	7
	Foreign	ministry	spokesman	Shen	Guofang	told	Reuters
gold-standard	B-ORG	I-ORG	0	B-PER	I-PER	0	B-ORG
predicted	0	0	0	B-PER	I-PER	0	B-PER

- Gold-standard entity spans:

(1, 2, ORG)

(4, 5, PER)

(7, 7, ORG)

false negative

- Predicted entity spans:

(4, 5, PER)

(7, 7, PER)

false positive

true positive

Challenges for evaluation

	1	2	3	4	5	6
	First	Bank	of	Chicago	announced	earnings
gold-standard	B-ORG	I-ORG	I-ORG	I-ORG	0	0
predicted	0	B-ORG	I-ORG	I-ORG	0	0

- In tuple notation:

gold-standard $(1, 4, \text{ORG}) \neq (2, 4, \text{ORG})$ predicted

- The example above creates **both** a **false negative** and a **false positive**.

Important concepts

- named entity recognition
- common entity types
- BIO tagging scheme
- gazetteer
- span-based precision/recall/F1-score

Linking Entities with Knowledge Bases




Knowledge bases

Definition

A knowledge base (KB) stores **structured and unstructured** information in a **machine-readable** way.

- Knowledge bases are often based on an explicit **object model**.
 - type hierarchy
 - ontology
- This contrasts with e.g. standard relational databases.
 - focused on storage & representing tabular data

WordNet

-  **WordNet** is a lexical database for English.
- Two main features:
 1. Words are mapped to sets of **cognitive synonyms** called “**synsets.**”
 2. Synsets are linked together with **semantic relations.**
- Version 3.1 was released in 2011 and contains **117,000** synsets.
 - However, there are no plans to update or maintain it.

WordNet: Synset examples

Noun

- **school** (an educational institution) *“the school was founded in 1900”*
- **school, schoolhouse** (a building where young people receive education) *“the school was built in 1932”; “he walked to school every morning”*
- **school, shoal** (a large group of fish) *“a school of small glittering fish swam by”*

Verb

- **school** (educate in or as if in school) *“the children are schooled in private institutions”*
- **educate, school, train, cultivate, civilize, civilise** (teach or refine to be discriminative in taste or judgment) *“she is well schooled in poetry”*

Source: [WordNet](#)

Semantic relations in WordNet

- **Synonymy**

two senses are (nearly) identical



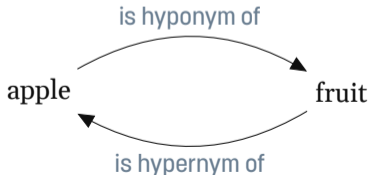
- **Antonymy**

two senses are opposites of each other



- **Hyponymy**

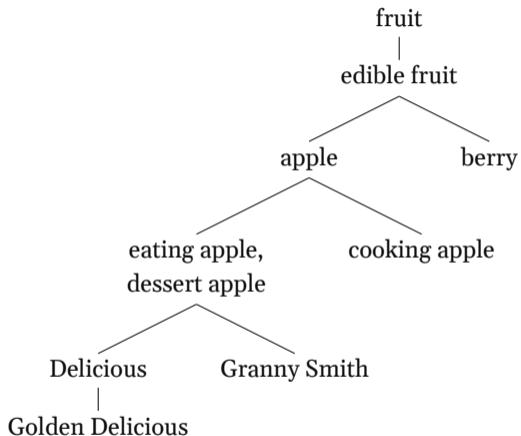
a sense is *more* specific than the other



- **Hypernymy**


a sense is *less* specific than the other

Hypernymy relations in WordNet (*a small excerpt*)

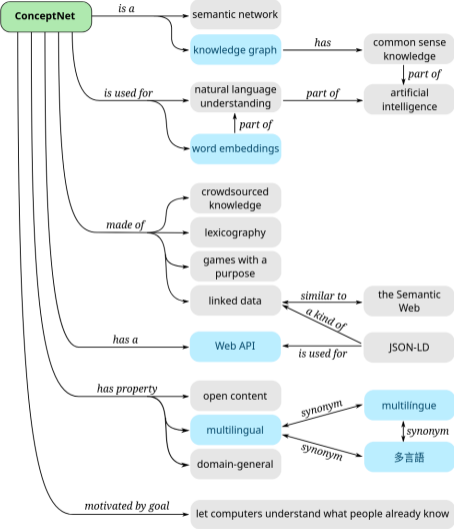


Source: [WordNet](#)

ConceptNet

-  **ConceptNet** is a multilingual knowledge graph.
- Contains “words and phrases and **common-sense relationships** between them.”
 - Contrast to WordNet: no concept of “synsets”
- Ten “core” languages with a combined vocabulary of **9.5 million** entries.
 - 68 additional “common” languages, plus 224 “rare” ones.
 - Combination of crowd-sourced (*e.g.* Wikipedia) and expert (*e.g.* WordNet) resources

ConceptNet visualized



Source: ConceptNet

ConceptNet: Edges for the English term "apple" (examples)

_ is a type of:

- en fruit
- en edible fruit
- en computer brand
- en friut

is a type of _:

- en cooking apple
- en crab apple
- en gala apple
- en pippin

_ is used for:

- en eating
- en making apple pie
- en computing
- en dessert

_ is at location:

- en apple tree
- en a grocery store
- en the fridge
- en a horses mouth

synonyms:

- es manzana
- fr pomme
- ja 苹果
- sv äpple

derived terms:

- en apple blossom
- en apple juice
- en apple pie
- en bad apple


properties:

- en red
- en green
- en opaque

symbols:

- mul 
- mul 

DBpedia

-  **DBpedia** is a crowd-sourced open knowledge graph.
 - Structured content extracted from “various **Wikimedia** projects”
- Ontology with **4.2 million** instances and 3000 different properties.
- Can be downloaded or queried through APIs.
 - **SPARQL**: standard query language for semantic databases

The Lord of the Rings

property	value
dbo:author	dbr:J. R. R. Tolkien
dbo:literaryGenre	dbr:Adventure novel dbr:High fantasy
dbo:previousWork	dbr:The Hobbit
dbo:publicationDate	1954-07-29
dbo:publisher	dbr:Allen & Unwin
dbo:wikiPageID	29798
dbp:language	English

Querying DBpedia with SPARQL (I)

- Example query: “*musicians who were born in Linköping*”

```
1 SELECT ?name ?birth ?description ?person WHERE {
2     ?person a dbo:MusicalArtist .
3     ?person dbo:birthPlace :Linköping .
4     ?person dbo:birthDate ?birth .
5     ?person foaf:name ?name .
6     ?person rdfs:comment ?description .
7     FILTER ( LANG(?description) = 'en' ) .
8 }
```

Query in SPARQL Explorer

Querying DBpedia with SPARQL (II)

name	birth	description	person
Jonna Lee	1981-10-03	Jonna Emily Lee (born 3 October 1981) is a Swedish singer, songwriter, record producer and visual director. Lee is the ...	🔗 Jonna_Lee_(singer)
Louise Hoffsten	1965-11-06	Louise Hoffsten, born September 6, 1965 in Linköping, Sweden, is a Swedish songwriter, musician and singer ...	🔗 Louise_Hoffsten
Ludwig Göransson	1984-09-01	Ludwig Emil Tomas Göransson [...] is a Swedish composer, conductor and record producer. He has scored films such as ...	🔗 Ludwig_Göransson
Martin Axenrot	1979-03-05	Erik Martin “Axe” Axenrot (born 5 March 1979 in Linköping, Sweden) is a Swedish death metal drummer, best known as the former drummer for progressive death metal band Opeth from 2006 to 2021.	🔗 Martin_Axenrot

Querying DBpedia with SPARQL (III)

- Linked knowledge graphs such as DBpedia allow for very detailed queries:

“soccer players who were born in a country with more than 10 million inhabitants, who played as goalkeeper with the jersey number 1, who played for a club that has a stadium with more than 30,000 seats, and whose club country is/was different from their birth country”

→ *See this [query and its results](#) in SPARQL Explorer.*

Linking entities to knowledge bases

- Wikipedia's [disambiguation page for "Washington"](#) lists over 120 entries.

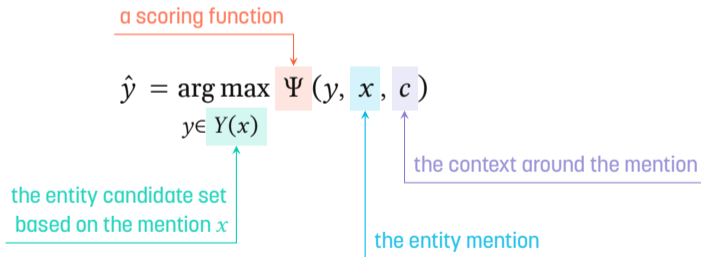
Washington was born into slavery.

George_Washington	Washington_Land	University_of_Washington
Washington_(state)	Washington_Avenue	Washington_College
Washington,_D.C.	Washington_Boulevard	Washington_School
Washington_(musician)	Washington_County	SS_Washington
Washington_(footballer,_born_1953)	Washington_Island	USS_Washington
Washington_(footballer,_born_1985)	Washington_Park	Washington_Capitals
Washington_(footballer,_born_1989)	Washington_Square	Washington_Huskies
F.C._Washington	Washington_Township	Washington_Wizards

A small sample of Wikipedia page titles matching "Washington"

Entity linking as candidate ranking

1. Identify a **set of candidate entities** for the entity mention.
 - e.g. in Wikipedia: all pages containing the mention in the title
2. **Rank** the candidates and pick the **highest-ranked** one.



Adapted from Eisenstein (2019)

Candidate ranking with a set of classifiers

- Both the candidate set and the ranking depend on the mention x .
- One way to implement this is to build **one classifier per mention**.
 - Naive Bayes, logistic regression, SVM, ...
- Each classifier **predicts the entity** given the context.
 - classes correspond to the candidate set for the respective mention

$$\hat{y} = \arg \max_{y \in Y(x)} P_x(y|c)$$

the entity candidate set depends on the mention x

the probability distribution depends on the mention x

Important concepts

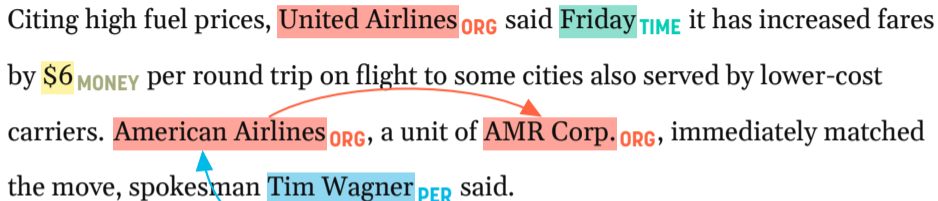
- knowledge bases/graphs
- WordNet
- ConceptNet
- DBpedia
- entity linking
- candidate ranking

Relation Extraction



Semantic relations: Example

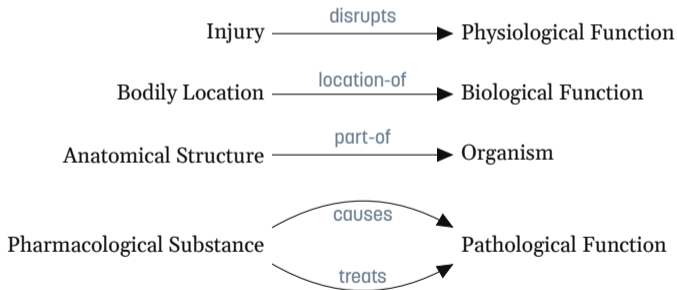
Citing high fuel prices, **United Airlines**_{ORG} said **Friday**_{TIME} it has increased fares by **\$6**_{MONEY} per round trip on flight to some cities also served by lower-cost carriers. **American Airlines**_{ORG}, a unit of **AMR Corp.**_{ORG}, immediately matched the move, spokesman **Tim Wagner**_{PER} said.



Tim Wagner $\xrightarrow{\text{is spokesman of}}$ American Airlines $\xrightarrow{\text{is unit of}}$ AMR Corp.

Relations in UMLS

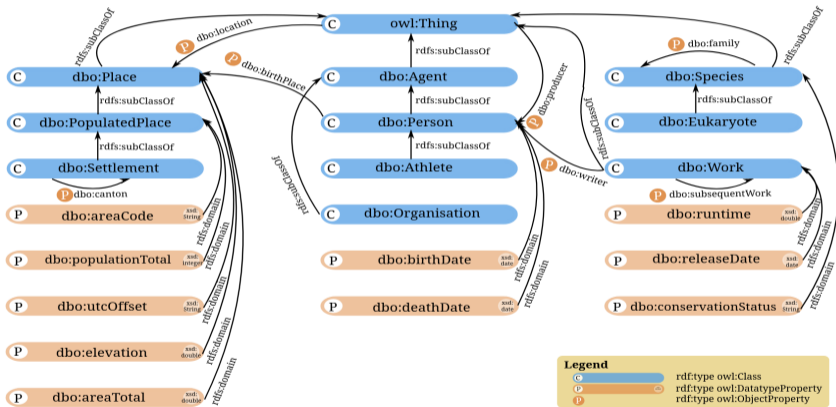
- The **Unified Medical Language System (UMLS)** defines entities and relations for the medical domain.
 - 134 broad categories, 54 relation types



Adapted from Jurafsky & Martin (2023)

Relations in DBpedia

- The **DBpedia ontology** defines over 1,300 classes with over 2,500 properties.



Source: Lehmann et al. (2012)

Relation extraction with regular expressions

- Simple relation extraction can be done with **regular expressions**.

Example: `.*\bborn.*\b`

- **August Strindberg**_{PER}, born **January 22, 1849**_{DATE} ...
→ \1 has-birth-date \2
- **August Strindberg**_{PER}, who was born in **1849**_{DATE} ...
→ \1 has-birth-date \2

Text patterns for the “X is-a Y” relation

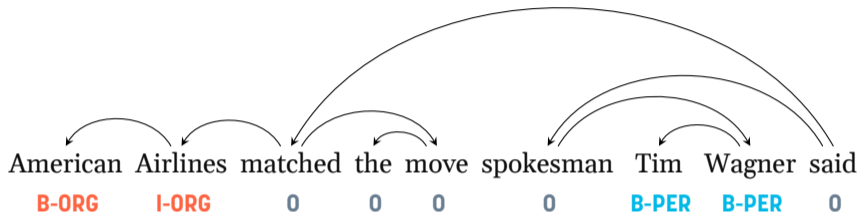
pattern	example
X and other Y	... temples, treasuries, and other civic buildings.
X or other Y	Bruises, wounds, broken bones or other injuries ...
Y such as X	The bow lute, such as the Bambara ndang ...
such Y as X	... such authors as Herrick, Goldsmith, and Shakespeare.
Y including X	... common-law countries, including Canada.
Y, especially X	European countries, especially France and Spain, ...

Adapted from Jurafsky & Martin (2017)

Relation extraction with dependency trees

1. Perform **named entity recognition** on the input.
2. Perform **dependency parsing** on the input.
3. For each pair of named entities, extract the **shortest path** between them in the dependency tree.
4. Let a classifier **predict the relation type** based on the extracted features.
 - can take all features from step 1–3 as input
 - prediction can be “none” if there is no relation

Relation extraction with dependency trees: Example (I)



- Extracted path between the two entities:

American Airlines_{ORG} ← matched ← said → spokesman → Tim Wagner_{PER}

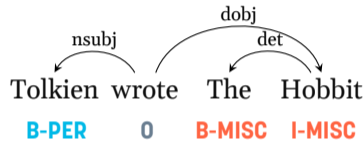
Relation extraction with dependency trees: Example (II)



- Arthur Conan Doyle_{PER} ← born → in → 1859_{DATE}
- Arthur Conan Doyle_{PER} ← born → in → Edinburgh, Scotland_{LOC}
- 1859_{DATE} ← in ← born → in → Edinburgh, Scotland_{LOC}

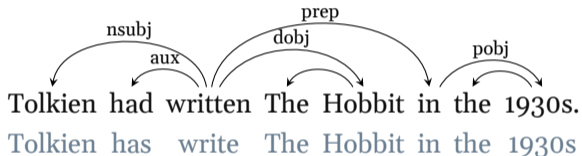
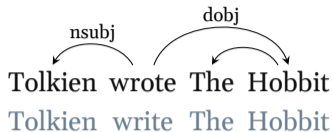
Using dependency arcs with labels

- Dependency parsers typically output both **arcs** and **arc labels**.
- Arc labels indicate **syntactic relationships** between the edges.
 - NSUBJ: nominal subject
 - DOBJ: direct object

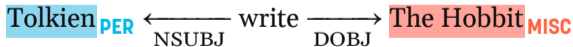


- Arc labels can be used to further narrow down the semantic relation between entities.

Relation extraction with arc labels: Example



- With lemmatization, both sentences produce exactly the same path:



Important concepts

- relation extraction
- regular expressions
- dependency trees
- dependency arcs & arc labels

